Milad Mikeal

CS 162

10/10/18

Project 2

Design:

- Create animal class
  - Create necessary variables
    - Age, cost, numBabies, baseFoodCost, and payoff
  - Create getter and setter methods
- Create Tiger class
  - Inherit from animal class
- Create Penguin class
  - Inherit from animal class
- Create turtle class
  - Inherit from animal class
- Create Zoo class
  - Create dynamically allocated array for each animal
  - Have a create method for each animal
    - Add each new animal to necessary array
    - Double size of array if more space needed
  - Have Boolean method for each animal to check if there are any animals old enough to have babies
  - Have delete Boolean methods to check if there is an animal of such type to delete
  - Random event method
    - If 1
      - Use rand to determine which animal gets sick and dies
      - If animal not available choose different animal
    - If 2
      - Use rand to determine bonus payoff for tigers
    - If 3
      - Use rand to determine which animal will have babies
      - Check if the animal chosen has any adults
      - Create necessary amount of babies
    - If 4: nothing
  - Calculate profit method
    - Loop through each animal array and calculate profit for each day
    - Add profit to bank
    - Reset tigers payoff back to default
      - In case random event was boom in attendance

    - Increment Age method
        - Loop through each animal array and increment age by 1
        - Subtract food cost for each animal
- Main
    - Create menu object from menu class (main menu)
        - Prompt user to start or quit
    - If user selected to start
        - Create Zoo object
        - Declare variables
        - Create menu to determine starting animals
        - Have user select 1 or 2 of each animal to start
        - Create loop to run until user quits/run out of money
            - Display day and balance
            - Call increment age method
            - Call randomevent method
            - Call calcprofit method
            - Display days profit
            - Ask user if they would like to buy new adult animal
            - Ask user if they would like to keep playing

Test Table: Main Menu – 1 for start, 2 for quit

Test Case	**Input**	**Expected Outcome**	**Outcome**
Input too low	Input < 1	Prompt user to re-enter value	User prompted to re-enter value
Input too high	Input > 2	Prompt user to re-enter value	User prompted to re-enter value
Input in range	Input = 1 or Input = 2	Proceed accordingly depending on user's input	Program proceeded accordingly
Non-numeric input	Input = alpha	Prompt user to re-enter value	User prompted to re-enter value

Test Table: Starting Animal

Test Case	Input	Expected Outcome	Outcome
Input too low	Input < 1	Prompt user to re-enter value	User prompted to re-enter value
Input too high	Input > 4	Prompt user to re-enter value	User prompted to re-enter value
Input in range	Input 1 - 4	Proceed accordingly depending on user's input	Program proceeded accordingly
Non-numeric input	Input = alpha	Prompt user to re-enter value	User prompted to re-enter value

Test Table: Buy Adult Animal – Yes/No

Test Case	Input	Expected Outcome	Outcome
Input too low	Input < 1	Prompt user to re-enter value	User prompted to re-enter value
Input too high	Input > 2	Prompt user to re-enter value	User prompted to re-enter value
Input in range	Input = 1 or Input = 2	Proceed accordingly depending on user's input	Program proceeded accordingly
Non-numeric input	Input = alpha	Prompt user to re-enter value	User prompted to re-enter value

Test Table: Choose Adult Animal

Test Case	Input	Expected Outcome	Outcome
Input too low	Input < 1	Prompt user to re-enter value	User prompted to re-enter value
Input too high	Input > 4	Prompt user to re-enter value	User prompted to re-enter value
Input in range	Input 1 - 4	Proceed accordingly depending on user's input	Program proceeded accordingly
Non-numeric input	Input = alpha	Prompt user to re-enter value	User prompted to re-enter value

Test Table: Keep Playing – Yes/No

Test Case	Input	Expected Outcome	Outcome
Input too low	Input < 1	Prompt user to re-enter value	User prompted to re-enter value
Input too high	Input > 2	Prompt user to re-enter value	User prompted to re-enter value
Input in range	Input = 1 or Input = 2	Proceed accordingly depending on user's input	Program proceeded accordingly
Non-numeric input	Input = alpha	Prompt user to re-enter value	User prompted to re-enter value

Reflection:

I thought this project was certainly more difficult than project 1. The concepts were just a bit more difficult. Project 1 dealt with material from CS161. This project dealt with inheritance, dynamic memory allocation in classes, pointers, etc. Furthermore, it was just a lot more work.

Although it was more difficult than project 1, it didn't think it was extremely difficult. It was certainly tedious though. I had to account for lots of situations. For example, in the case of an animal getting sick and dying, I had to set up a conditional to check if that animal existed in the array, and if it didn't, I had to move onto the next animal. Then I had to do the same for that animal. Same thing with babies being born. If there were no adults, I had to check for the next animal.

The most difficult part for me was dealing with the dynamically allocated arrays as member variables of the Zoo class. More specifically, after creating say a Tiger object, how to add it to the array. However, once that part was figured out everything else just came together, piece by piece. All in all, I actually enjoyed the project, and I feel like I learned a lot. I find myself really enjoying the difficulty of the assignments of this class. After each assignment, I feel like I learned so much.